Foundation of Cryptography, Lecture 1
One-Way Functions

Handout Mode

Iftach Haitner, Tel Aviv University

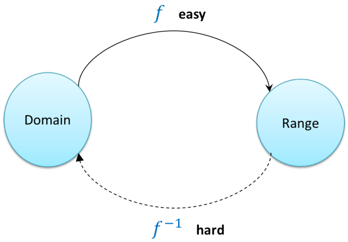
Tel Aviv University.

Feb. 18-25, 2014

Section 1

One-Way Functions

Informal discussion



A one-way function (OWF) is:

- Easy to compute, everywhere
- Hard to invert, on the average

- Why should we care about OWFs?
- Hidden in (almost) any cryptographic primitive: necessary for "cryptography"
- Sufficient for many cryptographic primitives

Formal definition

Definition 1 (one-way functions (OWFs))

A polynomial-time computable function $f\colon \{0,1\}^* \mapsto \{0,1\}^*$ is one-way, if

$$\Pr_{x \xleftarrow{R} \{0,1\}^n} \left[\mathsf{A}(1^n, f(x)) \in f^{-1}(f(x)) \right] = \operatorname{neg}(n)$$

for any PPT A.

- polynomial-time computable: there exists polynomial-time algorithm F, such that $F(x) = f(x)$ for every $x \in \{0,1\}^*$.
- neg: a function $\mu\colon \mathbb{N} \mapsto [0,1]$ is a negligible function of n, denoted $\mu(n) = \operatorname{neg}(n)$, if for any $p \in \operatorname{poly}$ there exists $n' \in \mathbb{N}$ such that $\mu(n) < 1/p(n)$ for all $n > n'$
- $x \xleftarrow{R} \{0,1\}^n$: x is uniformly drawn from $\{0,1\}^n$
- PPT: probabilistic polynomial-time algorithm.

We typically omit 1^n from the input list of A

Formal definition cont.

1. Is this the right definition?
   - Asymptotic
   - Efficiently computable
   - On the average
   - Only against PPT's
2. OWF $\implies \mathcal{P} \neq \mathcal{NP}$?
3. (most) Crypto implies OWFs
4. Do OWFs imply Crypto?
5. Where do we find them?
6. Non uniform OWFs

Definition 2 (Non-uniform OWF))

A polynomial-time computable function $f : \{0,1\}^* \mapsto \{0,1\}^*$ is non-uniformly one-way, if

$$\Pr_{x \leftarrow \{0,1\}^n}\left[C_n(f(x)) \in f^{-1}(f(x))\right] = \operatorname{neg}(n)$$

for any polynomial-size family of circuits $\{C_n\}_{n\in\mathbb{N}}$.

Length-preserving functions

Definition 3 (length preserving functions)

A function $f\colon \{0,1\}^* \mapsto f\colon \{0,1\}^*$ is length preserving, if $|f(x)| = |x|$ for every $x \in \{0,1\}^*$

Theorem 4

Assume that OWFs exit, then there exist length-preserving OWFs

Proof idea: use the assumed OWF to create a length preserving one

Partial domain functions

Definition 5 (Partial domain functions)

For $m, \ell \colon \mathbb{N} \mapsto \mathbb{N}$, let $h \colon \{0,1\}^{m(n)} \mapsto \{0,1\}^{\ell(n)}$ denote a function defined over input lengths in $\{m(n)\}_{n \in \mathbb{N}}$, and maps strings of length $m(n)$ to strings of length $\ell(n)$.

The definition of one-wayness naturally extends to such functions.

OWFs imply length-preserving OWFs cont.

Let $f\colon \{0,1\}^* \mapsto \{0,1\}^*$ be a OWF, let $p \in$ poly be a bound on its computing-time and assume wlg. that p is monotony increasing (can we?).

Construction 6 (the length preserving function)

Define $g\colon \{0,1\}^{p(n)} \mapsto \{0,1\}^{p(n)}$ as

$$g(x) = f(x_{1,\ldots,n}), 0^{p(n)-|f(x_{1,\ldots,n})|}$$

Note that g is well defined, length preserving and efficient (why?).

Claim 7

g is one-way.

How can we prove that g is one-way?
Answer: using reduction.

Proving that g is one-way

Proof:
Assume that g is not one-way. Namely, there exists PPT A, $q \in$ poly and infinite set $\mathcal{I} \subseteq \{p(n) \colon n \in \mathbb{N}\}$, with

$$\Pr_{x \leftarrow \{0,1\}^n} \left[\mathsf{A}(y) \in g^{-1}(g(x)) \right] > 1/q(n) \tag{1}$$

for every $n \in \mathcal{I}$.

We show how to use A for inverting f.

Algorithm 8 (The inverter B)

Input: 1^n and $y \in \{0,1\}^*$

1. Let $x = \mathsf{A}(1^{p(n)}, y, 0^{p(n)-|y|})$
2. Return $x_{1,\ldots,n}$

Claim 9

Let $\mathcal{I}' := \{n \in \mathbb{N} \colon p(n) \in \mathcal{I}\}$. Then

1. $\mathcal{I}'$ is infinite
2. $\Pr_{x \leftarrow \{0,1\}^n}[\mathsf{B}(1^n, f(x)) \in f^{-1}(f(x))] > 1/q(p(n))$ for every $n \in \mathcal{I}'$

This contradict the assumed one-wayness of f. $\square$

Proof: (1) is clear, (2)

$$\begin{aligned}
&\Pr_{x \leftarrow \{0,1\}^n}[\mathsf{B}(1^n, f(x)) \in f^{-1}(f(x))] \\
&= \Pr_{x \leftarrow \{0,1\}^n}[\mathsf{A}(1^{p(n)}, f(x), 0^{p(n)-n})_{1,\ldots,n} \in f^{-1}(f(x))] \\
&\geq \Pr_{x' \leftarrow \{0,1\}^{p(n)}}[\mathsf{A}(1^{p(n)}, g(x)) \in g^{-1}(g(x))] \geq 1/q(p(n))
\end{aligned}$$

From partial-domain OWFs to OWFs

Construction 10

Given a function $f\colon \{0,1\}^{\ell(n)} \mapsto \{0,1\}^{\ell(n)}$, define $f_{\mathrm{all}}\colon \{0,1\}^* \mapsto \{0,1\}^*$ as

$$f_{\mathrm{all}}(x) = f(x_{1,\ldots,k}), 0^{n-k}$$

where $n = |x|$ and $k := \max\{\ell(n') \leq n\colon n' \in [n]\}$.

Clearly, f_{all} is length preserving defined for every input length, and efficient (i.e., poly-time computable) in case f and ℓ are.

Claim 11

Assume f and ℓ are efficiently computable, f is one-way, and ℓ satisfies $1 \leq \frac{\ell(n+1)}{\ell(n)} \leq p(n)$ for some $p \in \mathrm{poly}$, then f_{all} is one-way function.

Proof: ?

Few Remarks

More “security-preserving" reductions exits.

Convention for rest of the talk

Let $f\colon \{0,1\}^n \mapsto \{0,1\}^n$ be a one-way function.

Weak One Way Functions

Definition 12 (weak one-way functions)

A poly-time computable function $f\colon \{0,1\}^* \mapsto f\colon \{0,1\}^*$ is α-one-way, if

$$\Pr_{x \leftarrow \{0,1\}^n}\left[\mathsf{A}(1^n, f(x)) \in f^{-1}(f(x))\right] \leq \alpha(n)$$

for any PPT A and large enough $n \in \mathbb{N}$.

1. (strong) OWF according to Definition 1, are neg-one-way according to the above definition
2. Can we “amplify" weak OWF to strong ones?

Strong to Weak OWFs

Claim 13

Assume there exists OWFs, then there exist functions that are $\frac{2}{3}$-one-way, but not (strong) one-way

Proof: For a OWF f, let

$$g(x) = \begin{cases} (1, f(x)), & x_1 = 1; \\ 0, & \text{otherwise } (x_1 = 1). \end{cases}$$

Weak to Strong OWFs

Theorem 14 (weak to strong OWFs (Yao))

Assume there exist $(1-\delta)$-weak OWFs with $\delta(n) \geq 1/q(n)$ for some $q \in \mathrm{poly}$, then there exist (strong) one-way functions.

- Idea: parallel repetition (i.e., direct product): Consider $g(x_1,\ldots,x_t) = f(x_1),\ldots,f(x_t)$ for large enough t
- Motivation: if something is somewhat hard, than doing it many times is (very) hard
- But, is it really so?

  Consider matrix multiplication: Let $A \in \mathbb{R}^{n\times n}$ and $x \in \mathbb{R}^n$

  Computing Ax takes $\Theta(n^2)$ times, but computing $A(x_1,x_2,\ldots,x_n)$ takes ...only $O(n^{2.3\ldots}) < \Theta(n^3)$
- Fortunately, parallel repetition does amplify weak OWFs :-)

Amplification via Parallel Repetition

Theorem 15

Let $f\colon \{0,1\}^n \mapsto \{0,1\}^n$*, and for* $t(n) := \left\lceil \frac{\log^2 n}{\delta(n)} \right\rceil$ *define* $g\colon (\{0,1\}^n)^{t(n)} \mapsto (\{0,1\}^n)^{t(n)}$ *as*

$$g(x_1,\ldots,x_{t(n)}) = f(x_1),\ldots,f(x_{t(n)})$$

Assume f *is* $(1-\delta)$*-weak OWF and* $\delta(n) = 1/q(n)$ *for some (positive)* $q \in \mathrm{poly}$*, then* g *is a one-way function.*

Clearly g is efficient. Is it one-way? Proof via reduction: Assume $\exists$ PPT A violating the one-wayness of g, we show there exists a PPT B violating the weak hardness of f.

Difficultly: We need to use an inverter for g with low success probability, e.g., $\frac{1}{n}$, to get an inverter for f with high success probability, e.g., $\frac{1}{2}$ or even $1-\frac{1}{n}$

In the following we fix (an assumed) PPT A, $p \in \mathrm{poly}$ and infinite set $\mathcal{I} \subseteq \mathbb{N}$ s.t.

$$\Pr_{w \xleftarrow{R} \{0,1\}^{t(n)\cdot n}}[\mathsf{A}(g(w)) \in g^{-1}(g(w))] \geq 1/p(n)$$

for every $n \in \mathcal{I}$. We also “fix" $n \in \mathcal{I}$ and omit it from the notation.

Proving that g is One-Way – the Naive Approach

Assume A attacks each of the t outputs of g independently: $\exists$ PPT A' such that $\mathsf{A}(z_1,\ldots,z_t) = \mathsf{A}'(z_1)\ldots,\mathsf{A}'(z_t)$

It follows that A' inverts f with probability greater than $(1-\delta(n))$. Otherwise

$$\Pr_{w \xleftarrow{R} \{0,1\}^{t(n)\cdot n}}[\mathsf{A}(g(w)) \in g^{-1}(g(w))] = \prod_{i=1}^{t} \Pr_{x \xleftarrow{R} \{0,1\}^{n}}\left[\mathsf{A}'(f(x)) \in f^{-1}(f(x))\right]$$
$$\leq (1-\delta(n))^{t(n)} \leq e^{-\log^2 n} \leq n^{-\log n}$$

Hence A' violates the weak hardness of f

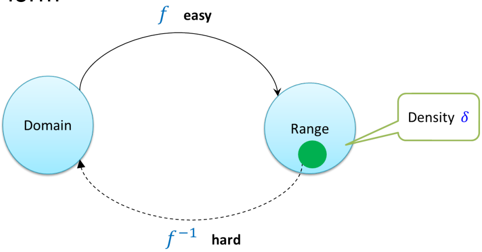
A less naive approach would be to assume that A goes over the inputs sequentially.

Unfortunately, we can assume none of the above.

Any idea?

Hardcore Sets

Assume f is of the form



Definition 16 (hardcore sets)

$\mathcal{S} = \{\mathcal{S}_n \subseteq \{0,1\}^n\}$ is a δ-hardcore set for $f\colon \{0,1\}^n \mapsto \{0,1\}^n$, if:

1. $\Pr_{x \xleftarrow{R} \{0,1\}^n}[f(x) \in \mathcal{S}] \geq \delta(n)$ for large enough n, and
2. For any PPT A and $q \in \text{poly}$: for large enough n, it holds that $\Pr\left[\mathsf{A}(y) \in f^{-1}(y)\right] \leq \frac{1}{q(n)}$ for every $y \in \mathcal{S}_n$.

Assuming f has a δ seems like a good starting point :-)

Unfortunately, we do not know how to prove that f has hardcore set :-<

Failing Sets

Definition 17 (failing sets)

A function $f\colon \{0,1\}^n \mapsto \{0,1\}^n$ has a δ-failing set for a pair (A, q) of algorithm and polynomial, if exists $\mathcal{S} = \{\mathcal{S}_n \subseteq \{0,1\}^n\}$, such that the following holds for large enough n:

1. $\Pr_{x \xleftarrow{R} \{0,1\}^n}[f(x) \in \mathcal{S}_n] \geq \delta(n)$, and
2. $\Pr\left[\mathsf{A}(y) \in f^{-1}(y)\right] \leq 1/q(n)$, for every $y \in \mathcal{S}_n$

Claim 18

Let f be a $(1-\delta)$-OWF, then f has a $\delta/2$-failing set, for any pair of PPT A and $q \in \mathrm{poly}$.

Proof: Assume $\exists$ PPT A and $q \in \mathrm{poly}$, such that for any $\mathcal{S} = \{\mathcal{S}_n \subseteq \{0,1\}^n\}$ at least one of the following holds:

1. $\Pr_{x \xleftarrow{R} \{0,1\}^n}[f(x) \in \mathcal{S}_n] < \delta(n)/2$ for infinitely many n's, or
2. For infinitely many n's: $\exists y \in \mathcal{S}_n$ with $\Pr\left[\mathsf{A}(y) \in f^{-1}(y)\right] \geq 1/q(n)$.

We'll use A to contradict the hardness of f.

Using A to Invert f

For $n \in \mathbb{N}$, let $\mathcal{S}_n := \{y \in \{0,1\}^n \colon \Pr\left[\mathsf{A}(y) \in f^{-1}(y)\right] < 1/q(n)\}$.

Claim 19

$\exists$ infinite $\mathcal{I} \subseteq \mathbb{N}$ with $\Pr_{x \xleftarrow{R} \{0,1\}^n}[f(x) \in \mathcal{S}_n] < \delta(n)/2$ for every $n \in \mathcal{I}$.

Algorithm 20 (The inverter B on input $y \in \{0,1\}^n$)

Do (with fresh randomness) for $n \cdot q(n)$ times:
If $x = \mathsf{A}(y) \in f^{-1}(y)$, return x

Clearly, B is a PPT

Claim 21

For $n \in \mathcal{I}$, it holds that $\Pr_{x \xleftarrow{R} \{0,1\}^n}\left[\mathsf{B}(f(x)) \in f^{-1}(f(x))\right] > 1 - \frac{\delta(n)}{2} - 2^{-n}$

Proof: ?

Hence, for large enough $n \in \mathcal{I}$: $\Pr_{x \xleftarrow{R} \{0,1\}^n}\left[\mathsf{B}(f(x)) \in f^{-1}(f(x))\right] > 1 - \delta(n)$.

Namely, f is not $(1-\delta)$-one-way $\square$

Proving g is One-Way cont.

We show that is g is not one way, then f has no $\delta/2$ flailing-set for some PPT B and $q \in$ poly.

Claim 22

Assume $\exists$ PPT A, $p \in$ poly and an infinite set $\mathcal{I} \subseteq \mathbb{N}$ such that

$$\Pr_{w \xleftarrow{R} \{0,1\}^{t(n)\cdot n}} \left[\mathsf{A}(g(x)) \in g^{-1}(g(w)) \right] \geq \frac{1}{p(n)}$$

for every $n \in \mathcal{I}$. Then $\exists$ PPT B such that

$$\Pr_{x \xleftarrow{R} \{0,1\}^n | y = f(x) \in \mathcal{S}_n} \left[\mathsf{B}(y) \in f^{-1}(y) \right] \geq \frac{1}{t(n)p(n)} - n^{-\log n}$$

for every $n \in \mathcal{I}$ and every $\mathcal{S}_n \subseteq \{0,1\}^n$ with $\Pr_{x \xleftarrow{R} \{0,1\}^n} [f(x) \in \mathcal{S}_n] \geq \delta(n)/2$.

Fix $\mathcal{S} = \{\mathcal{S}_n \subseteq \{0,1\}^n\}$. By Claim 22, for every $n \in \mathcal{I}$, either

- $\Pr_{x \xleftarrow{R} \{0,1\}^n} [f(x) \in \mathcal{S}_n] < \delta(n)/2$, or
- $\Pr_{x \xleftarrow{R} \{0,1\}^n | y = f(x) \in \mathcal{S}_n} \left[\mathsf{B}(y) \in f^{-1}(y) \right] \geq \frac{1}{t(n)p(n)} - n^{-\log n} \overset{(\text{for large enough } n \in \mathcal{I})}{\geq} \frac{1}{2t(n)p(n)}$

  $\overset{(\text{for large enough } n \in \mathcal{I})}{\Longrightarrow} \exists y \in \mathcal{S}_n$: $\Pr \left[\mathsf{B}(y) \in f^{-1}(y) \right] \geq \frac{1}{2t(n)p(n)}$

Namely, f has no $\delta/2$ failing set for $(\mathsf{B}, q = 2t(n)p(n))$

The No Failing-Set Algorithm

Algorithm 23 (Inverter B on input $y \in \{0,1\}^n$)

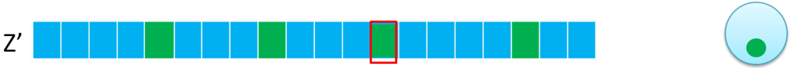
1. Choose $w \xleftarrow{R} (\{0,1\}^n)^{t(n)}$, $z = (z_1, \ldots, z_t) = g(w)$ and $i \xleftarrow{R} [t]$
2. Set $z' = (z_1, \ldots, z_{i-1}, y, z_{i+1}, \ldots, z_t)$
3. Return $\mathsf{A}(z')_i$

Fix $n \in \mathcal{I}$ and a set $\mathcal{S}_n \subseteq \{0,1\}^n$ with $\Pr_{x \xleftarrow{R} \{0,1\}^n}[f(x) \in \mathcal{S}] \geq \delta(n)/2$.

Claim 24

$$\Pr_{x \xleftarrow{R} \{0,1\}^n | y = f(x) \in \mathcal{S}_n}\left[\mathsf{B}(y) \in f^{-1}(y)\right] \geq \frac{1}{t(n) \cdot p(n)} - n^{-\log n}.$$

Proof: Assume for simplicity that A is deterministic.



Let $Typ = \{v \in \{0,1\}^{t(n) \cdot n} \colon \exists i \in [t(n)] \colon v_i \in \mathcal{S}_n\}$. $\Pr_z[Typ] \geq 1 - n^{-\log n}$.

For all $\mathcal{L} \subseteq \{0,1\}^{t(n) \cdot n}$: $\Pr_{z'}[\mathcal{L}] \geq \frac{\Pr_z[\mathcal{L} \cap Typ]}{t(n)} \geq \frac{\Pr_z[\mathcal{L}] - n^{-\log n}}{t(n)}$. $\square$

To conclude the proof take $\mathcal{L} = \{v \in \{0,1\}^{t(n) \cdot n} \colon \mathsf{A}(v) \in g^{-1}(v)\}$

Closing remarks

- Weak OWFs can be amplified into strong one

- Can we give a more security preserving amplification?
- Similar hardness amplification theorems for other cryptographic primitives (e.g., Captchas, general protocols)?
- What properties of the weak OWFs have we used in the proof?